IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

Application No.: 10/750,597	§	Examiner: Unknown
Filed: December 31, 2003	§	Group/Art Unit: Unknown
Inventors:	§	Atty. Dkt. No: 5760-14900/VRTS 0526
John Colgrove	§	
Par Botes	§	
Michael Timpanaro-Perrotta	§	
Charles Silvers	§	
Peter Vajgel	§	
Title: Multi-Class Storage Mechanism	§	

I hereby certify that this correspondence is being deposited with the United States Postal Service with sufficient postage as first class mail in an envelope addressed to Commissioner for Patents, P.O. Box 1450, Alexandria, VA 22313-1450, on the date indicated below.

Robert C. Kowert
Printed Name

Signature

March 26, 2004
Date

PETITION TO MAKE SPECIAL UNDER 37 C.F.R. § 1.102(d)

Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

Dear Sir:

This is a petition to make special under 37 C.F.R. § 1.102(d) and pursuant to MPEP 708.02(VIII). Please charge the petition fee under 37 C.F.R. § 1.17(h) to Meyertons, Hood, Kivlin, Kowert, & Goetzel, P.C. Deposit Account No. 501505/5760-14900/RCK.

A pre-examination search was performed by a search firm. The search firm reported that the field of search was class 711, subclasses 114, 136, and 165, and also included extensive computer searching in various U.S. and foreign patent related computer databases including the U.S. Patent and Trademark Office computer database. The search results returned to the undersigned by the search firm are nine U.S. patents or published patent application references, copies of which are included herewith. The

references are discussed in detail below, and the patentability of the claimed subject matter is pointed out.

Detailed Discussion of References

U.S. Patent 5,822,780 to Schutzman

Schutzman is generally concerned with hierarchical storage management for data base management systems. Particularly, Schutzman teaches a "hierarchical storage management system for database management systems that divides a database logically into separately managed regions, with each region being described by an entry in a vector kept in a regions file. The region entry contains a time stamp of the last time the region was accessed, the staging identifier of the region if it has been migrated, the base level backup staging identifier of the region if it has been baselined, and an indicator telling whether or not the region is resident online. Each managed region of the database is migrated to a separate staging file. When the database software issues a read or write input/output operation, the present invention sends the migration software a signal signifying this has occurred. The migration software of the present invention then updates the accessed time stamp, and checks to see if the region is resident. If it is not resident, it is staged in. The present invention creates and updates a migration candidate list ordered by date last accessed and region size. Both demand staging by the HSM, and user--initiated staging can then operate on the migration candidate list to migrate suitable files to tertiary storage." (Schutzman, abstract).

Schutzman further teaches: "...migration software 15 includes an HSM aware application 15a, such as backup, a stage-in daemon 15b, a master daemon 15c, a migration service process 15e, and an HSM file system monitor 15f. A standard application, such as database software 05, makes an I/O request 07 which is intercepted by interceptor 10, operating as part of a filesystem 08 in the kernel 09 of the operating system controlling computer system 00. As seen in FIG. 1c, callbacks 15cb can cause work requests 15wr to be sent to migration service process 15e, as appropriate. HSM file

system monitor 15f reads and writes data to local disks 20 through filesystem 08 as well, to satisfy migration requests." (Schutzman, column 5, line 66-column 6, line 11). Schutzman further teaches: "In a preferred embodiment, once a request has been intercepted, interceptor 10 creates an event that signifies to migration software 15 that an interception has occurred. Migration software 15 operates as part of an HSM system in a preferred embodiment and creates and uses a managed regions file 30 to keep track of activity on the database residing on local disks 20a through 20e. When migration software 15 is notified of the occurrence of the interception, it reads the vector entry in the managed regions file 30 that covers that region of the database on disk 20e specified by the I/O request." (Schutzman, column 5, lines 54-64). Schutzman further teaches: "...residency indicator 60 is set to 0, which indicates that the portion of the database which is the target of I/O request 07 refers to a portion of the database that has already been migrated to tertiary storage and is therefore not resident on disk 20e. At this point, migration software 15 will cause the HSM system of which it is a part, to bring the data covered by vector 50e from tertiary storage 40 to disk 20e." (Schutzman, column 6, lines 41-47).

Schutzman does not teach or suggest the combinations of features recited in each of claims 1-41 of the captioned application. For example, claim 1 recites a combination of features including a "multi-class storage mechanism configured to...migrate a portion of the data to different storage classes in the hierarchy of storage classes...wherein the migrated data remains online within the multi-class file system." Claims 2-13 depend from claim 1 and thus are not taught or suggested in Schutzman for at least the above reasons. Claim 14 recites a combination of features including "a multi-class storage mechanism configured to... migrate a portion of the data to different storage classes in the hierarchy of storage classes...wherein the migrated data remains online within the multi-class file system." Claim 15 recites a combination of features including "software means for assigning and migrating data to different storage classes in the hierarchy of storage classes according to a set of policies for the multi-class file system." Claim 16 recites a combination of features including "migrating a portion of the data to different storage classes in the hierarchy of storage classes...wherein the migrated data remains

online within the multi-class file system." Claims 17-29 depend from claim 16 and thus are not taught or suggested in Schutzman for at least the above reasons. Claim 30 recites a combination of features including "migrating a portion of the data to different storage classes in the hierarchy of storage classes...wherein the migrated data remains online within the multi-class file system." Claims 31-41 depend from claim 30 and thus are not taught or suggested in Schutzman for at least the above reasons.

U.S. Patent 6,311,252 to Raz

Raz is generally concerned with moving data between storage levels of a hierarchically arranged data storage system . Particularly, Raz teaches a "method of moving data between first, second, and third storage levels of a hierarchically arranged data storage system is described. The method includes the steps of dividing address space into a plurality of contiguous segments, each of which is further divided into a plurality of contiguous groups. Each of the groups may store a sequence of records, each of which is represented by a sequence of bytes. When it is necessary to move a designated record between the second and third levels of the data storage system, the segment in which that designated record is stored is moved; and when it is necessary to move the designated record between the first and second levels, the group in which that designated record is stored is moved." (Raz, abstract).

Raz further teaches: "FIG. 2 presents a schematic illustration of the organization of memory in the three level, hierarchical data storage system. The highest level of data storage, represented by line 100, is cache memory, which is divided into cache slots. The next lower level, represented by line 102 and implemented by disk arrays 16 shown in FIG. 1, is disk storage which is divided into logical volumes. The lowest level, represented by line 104, is tertiary storage, which in the described embodiment is implemented by tape drives 40, also shown in FIG. 1. It is divided into disk slots. The concept and role of slots can be more clearly understood by examining how a record is moved from one level of data storage to the next. For example, assume that a record A needs to be moved from the tape drive into the disk array from where it will be available to cache memory. To accomplish the transfer of record A, the controller first determines,

by looking at its local indexes, whether record A is located either in cache memory or in a logical volume in the disk array. Assuming that the record is in neither location, the controller then locates through an index within the tape system the physical location in tertiary storage of the disk slot that contains record A. Once having identified the physical location of the relevant disk slot, it then moves that disk slot into an available logical volume in the disk array and it updates index 50 (see FIG. 1) to indicate that the disk slot is now in the disk array." (Raz, column 5, lines 34-59). Raz further teaches: "With the block of data containing record A staged into disk memory, the controller then identifies the cache slot within the logical volume which contains record A and it moves that entire cache slot into available space in cache memory. That is, it either moves that data block including record A into an available cache slot, or it destages data from a cache slot in cache memory to its appropriate location in disk storage, to make an available cache slot in cache memory. The movement of data in the opposite direction, i.e., from cache memory to disk storage or from disk storage to tape storage, works very much the same way but in reverse. That is, when it is time to destage data from cache memory, the entire contents of the appropriate cache slot in cache memory are written back to the corresponding slot location within the appropriate disk volume, i.e., the disk volume that contains the data for that address range. Similarly, when destaging data from the disk array to the tape drive, disk slots are moved." (Raz, column 6, lines 3-21).

Raz does not teach or suggest the combinations of features recited in each of claims 1-41 of the captioned application. For example, claim 1 recites a combination of features including a "multi-class storage mechanism configured to… migrate a portion of the data to different storage classes in the hierarchy of storage classes in response to said application of the access information to the set of policies for the multi-class file system." Claims 2-13 depend from claim 1 and thus are not taught or suggested in Raz for at least the above reasons. Claim 14 recites a combination of features including a "multi-class storage mechanism configured to…migrate a portion of the data to different storage classes in the hierarchy of storage classes in response to said application of the access information to a set of policies for the multi-class file system." Claim 15 recites a combination of features including "software means for assigning and migrating data to

different storage classes in the hierarchy of storage classes according to a set of policies for the multi-class file system." Claim 16 recites a combination of features including "migrating a portion of the data to different storage classes in the hierarchy of storage classes in response to said applying the access information to the set of policies for the multi-class file system." Claims 17-29 depend from claim 16 and thus are not taught or suggested in Raz for at least the above reasons. Claim 30 recites a combination of features including "migrating a portion of the data to different storage classes in the hierarchy of storage classes in response to said applying the access information and other file information to the set of policies for the multi-class file system." Claims 31-41 depend from claim 30 and thus are not taught or suggested in Raz for at least the above reasons.

U.S. Patent Application Publication 2002/0069324 to Gerasimov et al. ("Gerasimov")

Gerasimov is generally concerned with a scalable storage architecture. Particularly, Gerasimov teaches a "Scalable Storage Architecture SSA system [that] integrates everything necessary for network storage and provides highly scalable and redundant storage space. The SSA comprises integrated and instantaneous back-up for maintaining data integrity in such a way as to make external backup unnecessary. The SSA also provides archiving and Hierarchical Storage Management (HSM) capabilities for storage and retrieval of historic data." (Gerasimov, abstract).

Gerasimov further teaches: "One aspect of the present invention is a redundant and scalable storage system for robust storage of data. The system includes a primary storage medium consisting of data and metadata storage, and a secondary storage medium. The primary storage medium has redundant storage elements that provide instantaneous backup of data stored thereon. Data stored on the primary storage medium is duplicated onto the secondary storage medium. Sets of metadata are stored in the metadata storage medium. Another aspect of the present invention is a method of robustly storing data using a system that has primary storage devices, secondary storage devices, and metadata storage devices. The method includes storing data redundantly on storage devices by duplicating it between primary and secondary devices. The method

also includes capabilities of removing data from the primary device and relying solely on secondary devices for such data retrieval thus freeing up primary storage space for other data." (Gerasimov, page 2, paragraphs 29-30).

Gerasimov further teaches: "Based upon data archiving policy, data that is most required and most required in a timely fashion are stored on the hard disks 14-26. As data ages further it is written to optical disks and stored in the optical disk library 30. Finally, for data that is older (for example, according to corporate data retention policies), it is subsequently moved to an 8-millimeter tape and stored in the tape library 32. The data archiving policies may be set by the individual company in convey to the operator of the present invention or certain default values for data storage are applied where data storage and retrieval policies are not specified." (Gerasimov, page 2, paragraphs 32-33).

Gerasimov does not teach or suggest the combinations of features recited in each of claims 1-41 of the captioned application. For example, claim 1 recites a combination of features including a "multi-class storage mechanism configured to...migrate a portion of the data to different storage classes in the hierarchy of storage classes in response to said application of the access information to the set of policies for the multi-class file system; wherein the migrated data remains online within the multi-class file system." Claims 2-13 depend from claim 1 and thus are not taught or suggested in Gerasimov for at least the above reasons. Claim 14 recites a combination of features including "a multi-class storage mechanism configured to...migrate a portion of the data to different storage classes in the hierarchy of storage classes in response to said application of the access information to a set of policies for the multi-class file system; wherein the migrated data remains online within the multi-class file system." Claim 15 recites a combination of features including "software means for assigning and migrating data to different storage classes in the hierarchy of storage classes according to a set of policies for the multi-class file system." Claim 16 recites a combination of features including "migrating a portion of the data to different storage classes in the hierarchy of storage classes in response to said applying the access information to the set of policies for the multi-class file system; wherein the migrated data remains online within the multi-class

file system." Claims 17-29 depend from claim 16 and thus are not taught or suggested in Gerasimov for at least the above reasons. Claim 30 recites a combination of features including "migrating a portion of the data to different storage classes in the hierarchy of storage classes in response to said applying the access information to the set of policies for the multi-class file system; wherein the migrated data remains online within the multi-class file system." Claims 31-41 depend from claim 30 and thus are not taught or suggested in Gerasimov for at least the above reasons.

U.S. Patent Application Publication 2002/0120763 to Miloushev et al. ("Miloushev")

Miloushev is generally concerned with file switch and switched file systems. Particularly, Miloushev teaches an "apparatus and method [that] are provided in a computer network to decouple the client from the server, by placing a transparent network node, also termed a file switch or file switch computer, between the client and the server. Usage of such a file switch allows reduced latency in file transfers, as well as scalable mirroring, striping, spillover, and other features." (Miloushev, abstract).

Miloushev further teaches: "The present invention...implement[s] a three-tiered architecture instead of the traditional client-server model. In this architecture, which we call a switched file system, an intermediate node, called a file switch, is introduced into the system between the network file clients and the network file servers. The introduction of a file switch into the system allows us to decouple clients and servers, thereby eliminating the one-to-one connections between them. As a result, clients are no longer aware which particular server handles a given file or even whether the file is handled by a single server or is distributed among multiple servers. This has major impact on the architecture and capabilities of the system. First, it makes it possible to aggregate the storage capacity of multiple servers in a manner transparent to clients. Second, it makes it possible to aggregate the performance of those servers by both distributing files among them and by striping individual files onto a plurality of servers that can perform in parallel. Third, it makes it possible to mirror individual files synchronously on multiple servers thereby increasing the availability of the data, again transparently to the clients. Fourth, by effectively aggregating the file servers in the

system into a single virtual network file system, the file switch greatly simplifies the management of the whole system to the point where it can be managed almost as easily as a single file server can. Moreover, since in a switched file system, the clients are effectively isolated from the particular process in which files, directories and data are distributed among file servers, it is possible to change that process at any time while the switched file system is operating. This is extremely important, since it constitutes a foundation for building adaptive, self-tuning network file systems that are capable of tracking the usage patterns and adjusting the distribution of data to optimize storage, performance and access." (Miloushev, page 8, paragraphs 116-119).

Miloushev does not teach or suggest the combinations of features recited in each of claims 1-41 of the captioned application. For example, claim 1 recites a combination of features including a "multi-class storage mechanism configured to...migrate a portion of the data to different storage classes in the hierarchy of storage classes in response to said application of the access information to the set of policies for the multi-class file system." Claims 2-13 depend from claim 1 and thus are not taught or suggested in Miloushev for at least the above reasons. Claim 14 recites a combination of features including "a multi-class storage mechanism configured to...migrate a portion of the data to different storage classes in the hierarchy of storage classes in response to said application of the access information to a set of policies for the multi-class file system." Claim 15 recites a combination of features including "software means for assigning and migrating data to different storage classes in the hierarchy of storage classes according to a set of policies for the multi-class file system." Claim 16 recites a combination of features including "migrating a portion of the data to different storage classes in the hierarchy of storage classes in response to said applying the access information to the set of policies for the multi-class file system." Claims 17-29 depend from claim 16 and thus are not taught or suggested in Miloushev for at least the above reasons. Claim 30 recites a combination of features including "migrating a portion of the data to different storage classes in the hierarchy of storage classes in response to said applying the access information to the set of policies for the multi-class file system." Claims 31-41 depend

from claim 30 and thus are not taught or suggested in Miloushev for at least the above reasons.

U.S. Patent Application Publication 2003/ 0023713 to Slater et al. ("Slater")

Slater is generally concerned with a monitoring appliance for data storage arrays, and a method of monitoring usage of said arrays. Particularly, Slater teaches a "monitoring appliance for a data storage array used by plural hosts to store data responds to stored metadata to interrogate the data storage array at intervals to establish the amount of usage of the data storage array. Each host can use the file system(s) and/or database(s) of its choice in portions of the data storage array allocated to it. The monitoring appliance has basic knowledge of all file systems/databases used by the hosts, and the metadata structure of those file systems/databases." (Slater, abstract).

Slater further teaches: "a disc array 10, which is the same as that in the prior art, is illustrated as linked to a monitoring appliance 12. The monitoring appliance 12 takes the form of a small computer running an embedded operating system, such as Linux, and performs only the monitoring function...The appliance 12 is however inaccessible to the hosts 1, 2 and 3 utilising the disc array 10." (Slater, page 2, paragraph 26). Slater further teaches: "With either manner of operation of the appliance 12, updates concerning the usage of the disc array 10 are posted to management station 16 via the management interface. The posting may conveniently use a simple kind of web publishing, such as an HTML web page, although any appropriate form that contains basic capacity usage information can be used. Appliance 12 posts such reports from time to time, e.g., at predetermined intervals, or as and when demanded by the management station." (Slater, page 3, paragraph 35). Slater further teaches: "The system and method enable the service provider to monitor in detail the usage of the disc array by the various hosts, both in terms of capacity used and in terms of the manner and timing of that usage." (Slater, page 3, paragraph 40).

Slater does not teach or suggest the combinations of features recited in each of claims 1-41 of the captioned application. For example, claim 1 recites a combination of

features including a "multi-class storage mechanism configured to…migrate a portion of the data to different storage classes in the hierarchy of storage classes." Claims 2-13 depend from claim 1 and thus are not taught or suggested in Slater for at least the above reasons. Claim 14 recites a combination of features including "a multi-class storage mechanism configured to…migrate a portion of the data to different storage classes in the hierarchy of storage classes." Claim 15 recites a combination of features including "software means for assigning and migrating data to different storage classes in the hierarchy of storage classes according to a set of policies for the multi-class file system." Claim 16 recites a combination of features including "migrating a portion of the data to different storage classes in the hierarchy of storage classes." Claims 17-29 depend from claim 16 and thus are not taught or suggested in Slater for at least the above reasons. Claim 30 recites a combination of features including "migrating a portion of the data to different storage classes in the hierarchy of storage classes." Claims 31-41 depend from claim 30 and thus are not taught or suggested in Slater for at least the above reasons.

U.S. Patent Application Publication 2003/0056069 to Cabrera et al. ("Cabrera")

Cabrera is generally concerned with buffering data in a hierarchical data storage environment. Particularly, Cabrera teaches a "system, a method, and program products for buffering data from a file in a hierarchical data storage system allocates data buffers and buffer management structures in memory to optimize performance of no recall requests. Buffer management structures, such as buffer headers and hash queue headers, are used to optimize performance of insert, search, and data buffer reuse operations. Buffer headers are managed in a least-recently-used queue in accordance with a relative availability status. Buffer headers are also organized in hash queue structures in accordance with file-based identifiers to facilitate searching for requested data in the buffers. Data buffers can be used to buffer different data blocks within the same file and can be recycled to buffer data from other data blocks and other files from the secondary storage device. Data in a data block may be reread by the requesting process or by other processes as long as the requested data remains valid. Lock fields are used to coordinate multi-thread and multi-user accesses." (Cabrera, abstract).

Cabrera further teaches: "An embodiment of the present invention buffers data from a file recorded on a secondary storage device in a hierarchical data storage system. Data-block-size data buffers for storing requested data blocks are allocated in memory and aligned to optimize the data transfer rate. Buffer headers are also allocated to facilitate management of the data buffers. The buffer headers are organized in a least-recently used (LRU) queue based on a relative availability status to coordinate the reuse of data buffers. The buffer headers are also organized in hash queue structures to optimize performance of insert and search operations. When a no recall request for data from a file recorded on a secondary storage device is received in association with a file-based identifier, the data buffers are searched first before an attempt to retrieve the data from the secondary storage device. If the data is already stored in a data buffer, the no recall request is serviced from the data buffer, instead of from the secondary storage device." (Cabrera, page 2, paragraph 21).

Cabrera does not teach or suggest the combinations of features recited in each of claims 1-41 of the captioned application. For example, claim 1 recites a combination of features including a "multi-class storage mechanism configured to...migrate a portion of the data to different storage classes in the hierarchy of storage classes in response to said application of the access information to the set of policies for the multi-class file system." Claims 2-13 depend from claim 1 and thus are not taught or suggested in Cabrera for at least the above reasons. Claim 14 recites a combination of features including "a multi-class storage mechanism configured to...migrate a portion of the data to different storage classes in the hierarchy of storage classes in response to said application of the access information to a set of policies for the multi-class file system." Claim 15 recites a combination of features including "software means for assigning and migrating data to different storage classes in the hierarchy of storage classes according to a set of policies for the multi-class file system." Claim 16 recites a combination of features including "migrating a portion of the data to different storage classes in the hierarchy of storage classes in response to said applying the access information to the set of policies for the multi-class file system." Claims 17-29 depend from claim 16 and thus are not taught or suggested in Cabrera for at least the above reasons. Claim 30 recites a combination of

features including "migrating a portion of the data to different storage classes in the hierarchy of storage classes in response to said applying the access information to the set of policies for the multi-class file system." Claims 31-41 depend from claim 30 and thus are not taught or suggested in Cabrera for at least the above reasons.

U.S. Patent Application Publication 2003/0065898 to Flamma et al. ("Flamma")

Flamma is generally concerned with managing object storage and retrieval in partitioned storage media. Particularly, Flamma teaches a "system for managing storage of an object in a storage system having a plurality of different storage media divided into different partitions, which includes a storage processor for determining whether a particular storage partition has reached a predetermined capacity threshold; a data migration processor for identifying within the particular storage partition an object to be moved and for identifying a target destination partition for the particular object in response to the capacity determination, the migration processor identifying the target destination partition based on one or more selected from the group consisting of (a) media type of the particular storage partition, and (b) information identifying related objects in the target destination partition; and a transfer processor for transferring data representing the particular object to the target destination partition." (Flamma, abstract).

Flamma further teaches: "The document-imaging system within which the present invention is to be used stores multimedia content in the form of documents...The document-imaging system organizes documents by filing them hierarchically into folders using a relational database." (Flamma, page 2, paragraphs 18-19). Flamma further teaches: "The present invention is directed to the hierarchical storage management subsystem of the above-described document-imaging system. It is capable of efficiently storing and retrieving data objects of various sizes and types on demand using different storage levels...the invention preferably supports multiple media types to balance acquisition and retrieval patterns with cost and performance. The storage processor of the invention preferably manages storage space by supporting multiple storage levels." (Flamma, pages 2-3, paragraph 26). Flamma further teaches: "The storage processor moves data objects between the storage levels to keep the current and most recently

accessed data on the fastest media available... the storage processor identifies within a particular storage partition an object to be moved based on frequency of access of the object relative to other objects in the storage partition, the object to be moved comprising an object with a lowest access frequency." (Flamma, page 3, paragraphs 27-28).

Flamma further teaches: "At some point, the user may need to retrieve a document(s) from a non-magnetic, long-term storage device. Such documents are then stored temporarily in a short-term storage area called a retrieval partition. Retrieval partition (614) is a field that allows the user to specify the partition that will be used for retrievals." (Flamma, page 9, paragraph 101).

Flamma does not teach or suggest the combinations of features recited in each of claims 1-41 of the captioned application. For example, claim 1 recites a combination of features including a "multi-class storage mechanism configured to...migrate a portion of the data to different storage classes in the hierarchy of storage classes...wherein the migrated data remains online within the multi-class file system." Claims 2-13 depend from claim 1 and thus are not taught or suggested in Flamma for at least the above reasons. Claim 14 recites a combination of features including "a multi-class storage mechanism configured to...migrate a portion of the data to different storage classes in the hierarchy of storage classes...wherein the migrated data remains online within the multi-class file system." Claim 15 recites a combination of features including "means for implementing a multi-class file system comprising a hierarchy of storage classes on a plurality of storage devices [and] software means for assigning and migrating data to different storage classes in the hierarchy of storage classes." Claim 16 recites a combination of features including "migrating a portion of the data to different storage classes in the hierarchy of storage classes...wherein the migrated data remains online within the multi-class file system." Claims 17-29 depend from claim 16 and thus are not taught or suggested in Flamma for at least the above reasons. Claim 30 recites a combination of features including "migrating a portion of the data to different storage classes in the hierarchy of storage classes...wherein the migrated data remains online

within the multi-class file system." Claims 31-41 depend from claim 30 and thus are not taught or suggested in Flamma for at least the above reasons.

U.S. Patent Application Publication 2003/0046270 to Leung et al. ("Leung")

Leung is generally concerned with techniques for storing data based upon storage policies. Particularly, Leung teaches a "Scalable Storage Architecture SSA system [that] integrates everything necessary for network storage and provides highly scalable and redundant storage space. The SSA comprises integrated and instantaneous back-up for maintaining data integrity in such a way as to make external backup unnecessary. The SSA also provides archiving and Hierarchical Storage Management (HSM) capabilities for storage and retrieval of historic data." (Leung, abstract).

Leung further teaches: "a data management system coupled to a heterogeneous data storage environment is configured to automate data management and storage functions. In this embodiment, the data management system is configured to monitor and analyze data and storage resource usage patterns and determine optimal storage locations for the data based upon the usage patterns. The data management system is also configured to determine storage locations for the data based upon characteristics of the data and the storage devices and based upon storage policies configured for the storage environment. The storage policies may be configured by a user (e.g., an end-user, a system administrator, a manager, etc.) of the storage environment." (Leung, page 2, paragraph 21). Leung further teaches: "DMS 104 determines locations for storing data in distributed network 100 based upon one or more storage policies configured for the storage environment, based upon information identifying characteristics of the data to be stored, and based upon information identifying characteristics of the storage devices available for storing the data in the storage environment." (Leung, page 3, paragraph 32).

Leung further teaches: "... storage devices may be characterized by the amount of time required to access data (referred to as "data access time") stored by the storage devices. For example, storage devices may be characterized as on-line storage devices, near-line storage devices, <u>off-line storage devices</u>, and others. The data access time for an

on-line storage device is generally shorter than the access time for a near-line storage device. The access time for an off-line storage is generally longer than the access time for a near-line storage device. An off-line storage device is generally a device that is not readily accessible to DMS 104. Examples of off-line storage devices include computer-readable storage media such as tapes, optical devices, and the like. User interaction may be required to access data from an off-line storage device. For example, if a tape is used as an off-line device, the user may have to make the tape accessible to DMS 104 before data stored on the tape can be restored by DMS 104." (Leung, page 4, paragraph 42).

Leung does not teach or suggest the combinations of features recited in each of claims 1-41 of the captioned application. For example, claim 1 recites a combination of features including a "multi-class storage mechanism configured to monitor access of data stored in a multi-class file system [and] migrate a portion of the data to different storage classes in the hierarchy of storage classes...wherein the migrated data remains online within the multi-class file system." Claims 2-13 depend from claim 1 and thus are not taught or suggested in Leung for at least the above reasons. Claim 14 recites a combination of features including "a multi-class storage mechanism configured to monitor access of data stored in the multi-class file system [and] migrate a portion of the data to different storage classes in the hierarchy of storage classes...wherein the migrated data remains online within the multi-class file system." Claim 15 recites a combination of features including "means for implementing a multi-class file system comprising a hierarchy of storage classes on a plurality of storage devices [and] software means for assigning and migrating data to different storage classes in the hierarchy of storage classes according to a set of policies for the multi-class file system." Claim 16 recites a combination of features including "monitoring access of data stored in a multi-class file system [and] migrating a portion of the data to different storage classes in the hierarchy of storage classes...wherein the migrated data remains online within the multi-class file system." Claims 17-29 depend from claim 16 and thus are not taught or suggested in Leung for at least the above reasons. Claim 30 recites a combination of features including "monitoring access of data stored in a multi-class file system [and] migrating a portion of the data to different storage classes in the hierarchy of storage classes in

response to said applying the access information to the set of policies for the multi-class file system; wherein the migrated data remains online within the multi-class file system." Claims 31-41 depend from claim 30 and thus are not taught or suggested in Leung for at least the above reasons.

U.S. Patent Application Publication 2004/0039891 to Leung et al. ("Leung")

Leung is generally concerned with optimizing storage capacity utilization based upon data storage costs. Particularly, Leung teaches "Techniques for optimizing capacity utilization among multiple storage units based upon costs associated with storing data on the storage units. Embodiments of the present invention automatically determine when data movement is needed to optimization storage utilization for a group of storage units. According to an embodiment of the present invention, in order to optimize storage utilization and storage cost, files are moved from a source storage unit to a target storage unit that has a lower data storage cost associated with it than the source storage unit. The storage units may be assigned to one or more servers." (Leung, abstract).

Leung further teaches: "SMS 110 is configured to provide storage management services for storage environment 100 according to an embodiment of the present invention. These management services include performing automated capacity management and data movement between the various storage units in the storage environment 100...SMS 110 is configured to monitor and gather information related to the capacity usage of the storage units in the storage environment and to perform capacity management (including managing capacity based upon data storage costs) and data movement based upon the gathered information. SMS 110 may perform monitoring in the background to determine the instantaneous state of each of the storage units in the storage environment. SMS 110 may also monitor the file system in order to collect information about the files such as file size information, access time information, file type information, etc. The information collected by SMS 110 may be stored in a memory or disk location accessible to SMS 110...The information stored in database 112 may include information 114 related to storage policies and rules configured for the storage environment, information 116 related to the various monitored storage units, information

118 related to the files stored in the storage environment, and other types of information 120...the stored information may be used to perform capacity management based upon data storage costs according to an embodiment of the present invention." (Leung, page 3, paragraphs 35-36).

Leung further teaches: "When file I/O driver module 908 determines that a requested file has been migrated from its original location to a different location, it may suspend the file open call and perform the following operations: (1) File I/O driver 908 may determine the actual location of the requested data file in storage environment 912. This can be done by looking up from the file header or stub file that is stored in the original location. Alternatively, if the file location information is stored in a persistent storage location (e.g., a database managed by PME module 904), file I/O driver 908 may determine the actual remote location of the file from that persistent location; (2) File I/O driver 908 may then restore the file content from the remote storage unit location; (3) File I/O driver 908 then resumes the file open call so that the application can resume with the restored data." (Leung, pages 10-11, paragraphs 110-111).

Leung does not teach or suggest the combinations of features recited in each of claims 1-41 of the captioned application. For example, claim 1 recites a combination of features including a "multi-class storage mechanism configured to monitor access of data stored in a multi-class file system [and] migrate a portion of the data to different storage classes in the hierarchy of storage classes...wherein the migrated data remains online within the multi-class file system." Claims 2-13 depend from claim 1 and thus are not taught or suggested in Leung for at least the above reasons. Claim 14 recites a combination of features including "a multi-class storage mechanism configured to monitor access of data stored in the multi-class file system [and] migrate a portion of the data to different storage classes in the hierarchy of storage classes...wherein the migrated data remains online within the multi-class file system." Claim 15 recites a combination of features including "means for implementing a multi-class file system comprising a hierarchy of storage classes on a plurality of storage devices [and] software means for assigning and migrating data to different storage classes in the hierarchy of storage

classes according to a set of policies for the multi-class file system." Claim 16 recites a combination of features including "monitoring access of data stored in a multi-class file system [and] migrating a portion of the data to different storage classes in the hierarchy of storage classes...wherein the migrated data remains online within the multi-class file system." Claims 17-29 depend from claim 16 and thus are not taught or suggested in Leung for at least the above reasons. Claim 30 recites a combination of features including "monitoring access of data stored in a multi-class file system [and] migrating a portion of the data to different storage classes in the hierarchy of storage classes in response to said applying the access information to the set of policies for the multi-class file system; wherein the migrated data remains online within the multi-class file system." Claims 31-41 depend from claim 30 and thus are not taught or suggested in Leung for at least the above reasons.

If any fees are due, the Commissioner is authorized to charge said fees to Meyertons, Hood, Kivlin, Kowert, & Goetzel, P.C. Deposit Account No. 501505/5760-14900/RCK.

Respectfully submitted,

Robert C. Kowert
Reg. No. 39,255
AGENT FOR APPLICANT(S)

Meyertons, Hood, Kivlin, Kowert, & Goetzel, P.C.
P.O. Box 398
Austin, TX 78767-0398
Phone: (512) 853-8850

Date: March 26, 2004